Introduction

Course Name : Introduction To Data Science

Faculty name : Dr. Ashraf Uddin

Section : [G]

ID	Name	Contribution
22-47196-1	SABBIR AHMED AL SEUM	Code Instruction, report writing [Graphical]
22-47125-1	AL FAHAD	English Version, Report writing [grammar]
22-46967-1	MD. NISHAT TASNIM	Report writing, Template selection
22-46888-1	RUBAYET ALAM AZAN	Data visualization , Formation , Grammar

Overview

Objective

To analyze English news articles from The Prothom Alo using text mining and visualization techniques, extract frequent words, and identify major themes through topic modeling.

Dataset

- **Source**: The Prothom Alo
- **Format**: CSV file (prothomalo_articles_with_text.csv)
- **Encoding**: UTF-8
- **Total Articles Analyzed**: [As per dataset]

Tools & Technologies

- **Programming Language**: R
- **IDE**: RStudio
- **Libraries Used**:

Library	**Purpose**
readr	Read CSV file
tm	Text preprocessing and corpus creation
dplyr	Data manipulation
tidytext	Tokenization
topicmodels	Topic modeling (LDA)
ggplot2	Plotting and visualization
wordcloud	Generate word clouds
RColorBrewer	Add color palettes

Gathering Links:

```
# Load links CSV (if you have it saved)
links_data <- read_csv("C:/Users/USER/Documents/prothomalo_links_with_labels.csv", show_col_types = FALSE)[,]

# Function to extract article details
extract_article_details <- function(url) {
  tryCatch({
    page <- read_html(url)
    closeAllConnections()
    Sys.sleep(1)
    #first making connection with the links of CSV then End the connection
    # Extract text from all <p> tags
    panel_nodes <- page %>% html_nodes("div.tabs-panel.is-active")
    if (length(panel_nodes) > 0) {
      xml2::xml_remove(panel_nodes)
    }
    date_published <- page %>%
      html_node("meta[itemprop='datePublished'], meta[property='article:published_time']") %>%
      html_attr("content")
    if (is.null(date_published)) date_published <- NA
    paragraphs <- page %>%
      html_nodes("p") %>%
      html_text() %>%
      paste(collapse = " ")  # Combine all text into one single string
    return(data.frame(
      article_text = paragraphs,
      date_published = date_published,
      stringsAsFactors = FALSE
    ))
  }, error = function(e) {
    message("Failed to scrape: ", url)
    return(data.frame(
      article_text = NA,
      stringsAsFactors = FALSE
    ))
  })
}

# Loop through all links and extract details
results <- links_data %>%
  mutate(full_url = paste0("", href)) %>%  # Assuming href is relative
  mutate(scraped = map(full_url, extract_article_details)) %>%
  unnest(scraped)
```

Term- Document Matrix:

```
# --- Topic Modeling (LDA) ---
num_topics <- 5
lda_model <- LDA(dtm, k = num_topics, control = list(seed = 1234))

cat("\nLDA topic modeling completed. Top terms per topic:\n")
library(tidytext)
library(tidyr)
library(tibble)

topic_terms <- tidy(lda_model, matrix = "beta")
top_terms <- topic_terms %>%
  group_by(topic) %>%
  slice_max(beta, n = 10) %>%
  ungroup() %>%
  arrange(topic, -beta)
print(top_terms)
```

Text Processing:

- Converted text to lowercase
- Removed punctuation, numbers, and extra whitespace
- Removed **English stopwords**
- Created a cleaned **corpus** for further analysis

```
# --- Step 2: Text Preprocessing ---
cat("Sample articles:\n")
print(head(data$article_text, 2))

corpus <- VCorpus(VectorSource(data$article_text))
```

Tokenization:

- Cleaned text was converted into individual **tokens (words)**
- Tokenized data stored for visualization and analysis

```
# --- Tokenization ---
text_df <- data.frame(text = sapply(corpus, as.character), stringsAsFactors = FALSE) %>%
  mutate(document = row_number())

tokens <- text_df %>%
  unnest_tokens(word, text)

cat("\nSample tokens:\n")
print(head(tokens, 20))
```

```
Sample tokens:
> print(head(tokens, 20))
         document        word
1               1        <NA>
2               2        <NA>
3               3        <NA>
4               4        <NA>
5               5        <NA>
6               6        <NA>
7               7        <NA>
8               8        <NA>
9               9        <NA>
10             10        <NA>
11             11        <NA>
12             12        <NA>
13             13        <NA>
14...14        14  bangladesh
14...15        14 nationalist
14...16        14       party
14...17        14    standing
14...18        14   committee
14...19        14      member
14...20        14        amir
```

Word Frequency Analysis:

- Constructed both **Term Document Matrix (TDM)** and **Document Term Matrix (DTM)**
- Identified most frequent words across all articles
- Saved word frequencies as CSV

```
# --- Word Frequency Analysis ---
tdm <- TermDocumentMatrix(corpus)
m <- as.matrix(tdm)
word_freqs <- sort(rowSums(m), decreasing = TRUE)
freq_df <- data.frame(word = names(word_freqs), freq = word_freqs)

cat("\nTop 20 most frequent words:\n")
print(head(freq_df, 20))
```

Data visualization:

```
# --- Bar Chart ---
bar_plot <- ggplot(freq_df[1:20, ], aes(x = reorder(word, freq), y = freq)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  labs(title = "Top 20 Most Frequent Words", x = "Word", y = "Frequency") +
  theme_minimal()
print(bar_plot)
ggsave(paste0(output_dir, "Top_20_Words_BarChart.png"), plot = bar_plot, width = 8, height = 6)

# --- Word Cloud ---
png(paste0(output_dir, "WordCloud.png"), width = 800, height = 600)
wordcloud(words = freq_df$word, freq = freq_df$freq, min.freq = 2, max.words = 100,
          random.order = FALSE, colors = brewer.pal(8, "Dark2"))
dev.off()

# --- Box Plot ---
box_plot <- ggplot(freq_df[1:50, ], aes(x = "", y = freq)) +
  geom_boxplot(fill = "orange") +
  labs(title = "Boxplot of Top 50 Word Frequencies", y = "Frequency") +
  theme_minimal()
print(box_plot)
ggsave(paste0(output_dir, "Boxplot_Frequencies.png"), plot = box_plot, width = 6, height = 4)
```

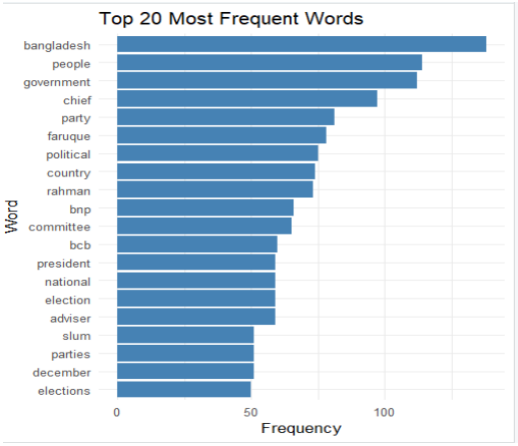
```
# --- Scatter Plot ---
scatter_plot <- ggplot(freq_df[1:50, ], aes(x = 1:50, y = freq)) +
  geom_point(color = "darkgreen") +
  labs(title = "Scatter Plot of Top 50 Word Frequencies", x = "Rank", y = "Frequency") +
  theme_minimal()
print(scatter_plot)
ggsave(paste0(output_dir, "ScatterPlot_Frequencies.png"), plot = scatter_plot, width = 6, height = 4)

# --- Point Plot with Labels ---
point_plot <- ggplot(freq_df[1:30, ], aes(x = reorder(word, freq), y = freq)) +
  geom_point(color = "purple", size = 3) +
  geom_text(aes(label = word), hjust = -0.2, size = 3) +
  coord_flip() +
  labs(title = "Point Plot of Top 30 Words", x = "Word", y = "Frequency") +
  theme_minimal()
print(point_plot)
ggsave(paste0(output_dir, "PointPlot_Frequencies.png"), plot = point_plot, width = 8, height = 6)
```

Visualizations:

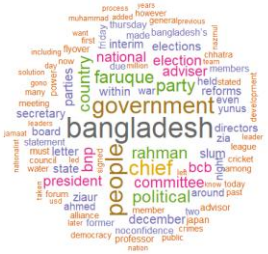
1. Bar Chart

- Top 20 most frequent words
- Horizontal bar plot saved as Top_20_Words_BarChart.png



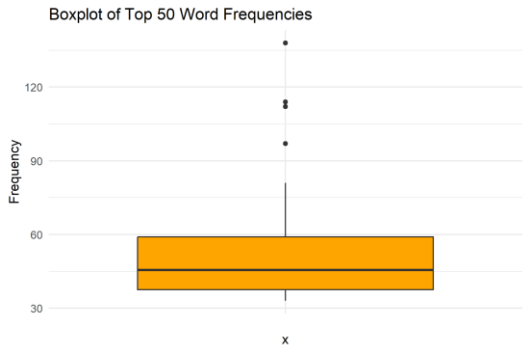
2. Word Cloud

- Up to 100 most frequent words (min.freq = 2)
- Visualized using Dark2 palette
- Saved as WordCloud.png



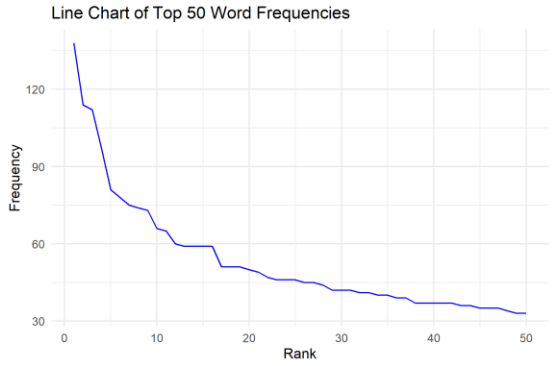
3. Box Plot

- Distribution of top 50 word frequencies
- Saved as Boxplot_Frequencies.png



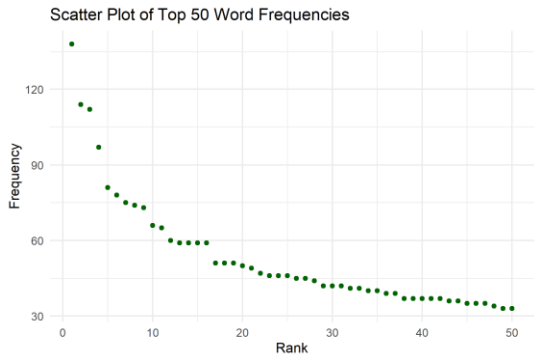
4. Line Chart

- Word frequency trend across top 50 words
- Saved as LineChart_Frequencies.png



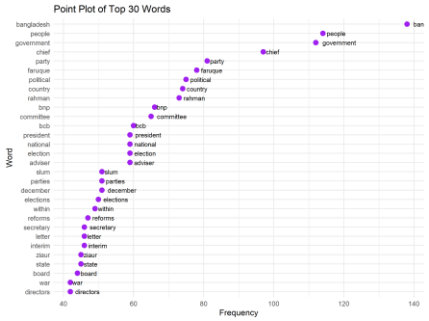
5. Scatter Plot

- Rank vs frequency for top 50 words
- Saved as ScatterPlot_Frequencies.png



6. Point Plot with Labels

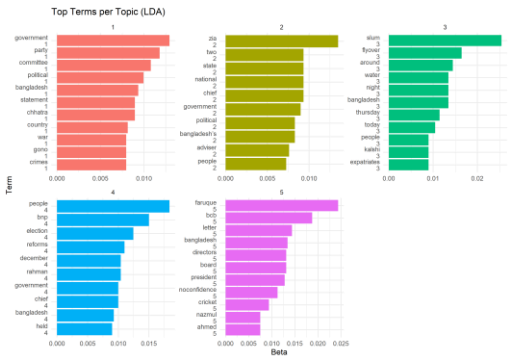
- Top 30 words plotted with direct labels
- Saved as PointPlot_Frequencies.png



Topic Modeling (LDA)

- Performed **Latent Dirichlet Allocation (LDA)** with 5 topics
- Extracted **top 10 words per topic**
- Bar chart created and saved as Top_Terms_Per_Topic.png

```
R 4.5.0 · 
> print(top_terms)
# A tibble: 53 × 3
   topic term          beta
   <int> <chr>        <dbl>
 1     1 government 0.0129
 2     1 party      0.0118
 3     1 committee  0.0108
 4     1 political  0.00994
 5     1 bangladesh 0.00933
 6     1 chhatra    0.00894
 7     1 statement  0.00894
 8     1 country    0.00813
 9     1 crimes     0.00795
10     1 gono       0.00795
# ℹ 43 more rows
# ℹ Use `print(n = ...)` to see more rows
>
```



Sample Output of Topics:

Topic	Top Words
1	women, commission, rights, groups, government
2	university, students, teachers, education
3	japan, japanese, yunus, ishiba, partnership
4	forest, acres, case, doe, political
5	rangamati, friday, landslides, road, teams